Scope Classification: An Instance-Based Learning Algorithm with a Rule-Based Characterisation

Nicolas Lachiche[1] and Pierre Marquis[2]

[1] LORIA, B.P. 239, 54506 Vandœuvre-lès-Nancy Cedex, France- e-mail: lachiche@loria.fr***
[2] CRIL/Université d'Artois, Rue de l'Université, S.P. 16, 62307 Lens Cedex, France - e-mail: marquis@cril.univ-artois.fr

Abstract. *Scope classification* is a new instance-based learning (IBL) technique with a rule-based characterisation. Within the scope approach, the classification of an object o is based on the examples that are closer to o than every example labelled with another class. In contrast to standard distance-based IBL classifiers, scope classification relies on partial pre-orderings $\leq_o$ between examples, indexed by objects. Interestingly, the notion of closeness to o that is used characterises the classes predicted by all the rules that cover o and are relevant and consistent for the training set. Accordingly, scope classification is an IBL technique with a rule-based characterisation. Since rules do not have to be explicitly generated, the scope approach applies to classification problems where the number of rules prevents them from being exhaustively computed.

1 Introduction

In this paper, we are interested in supervised learning. Given a *training set* T and a set of *objects* o to be classified, our goal is to derive a classifier that best approximates in the best way the *target function*, i.e., the function that maps every object to its right class. Because only the classes $class(e)$ of the training examples e (also called instances) are known, this classifier must be *induced* from examples. This requires some additional knowledge, that typically has the form of a *similarity assumption*. Such an assumption states that "every object belongs to the class of its nearest neighbours in the training set" or "every object shares the properties relevant to class membership that every example exhibits".

Instance-based learning (IBL) is based on a straightforward interpretation of the similarity assumption. In the simplest case, every object o is classified according to its nearest instance, according to some similarity measure or to some distance measure. The k-nearest neighbours of o can also be used; in this case, the class of o is computed as the majority class of its k nearest neighbours from T.

*** Currently at Department of Computer Science, University of Bristol, Merchant Venturers Building, Woodland Road, Bristol BS8 1UB, United Kingdom

Another approach to learning a classifier from examples is *rule-based classification*. A rule $c \rightarrow (y = v_y)$ is said to *classify* (to cover, or to be satisfied by) an object o if $o \models c$, i.e., c is a logical consequence of o. A rule $r_1 = c_1 \rightarrow (y = v_y)$ is said at least as *general* as a rule $r_2 = c_2 \rightarrow (y = v_y)$ if and only if $c_2 \models c_1$. Given a training set T, a rule $r = c \rightarrow (y = v_y)$ is *consistent* for T if and only if for every example e of T, if r classifies e then $class(e) = v_y$. A rule r is *relevant* for T if and only if r is satisfied by at least one example from T.

The number of relevant and consistent rules for a training set can be huge (exponential in the number of attributes). As a consequence, many rule induction algorithms only generate some of these rules, typically the most discriminating ones. Because the most discriminating rules are not always sufficient to approximate the target function in a satisfying way, SE-Learn [Rymon, 1993] completes them with the second most discriminating ones, and so on, until all the (most general) relevant and consistent rules for the training set are generated. [Rymon, 1996b] shows that SE-Learn is more robust to noise than decision trees.

In the following, a new approach to learning from examples, called *scope classification*, is introduced. The scope approach is a point where IBL and rule-based techniques meet. Roughly, the scope algorithm classifies every object o according to the examples of T that are "closer" to o than any example labelled with another class. Since every object can be classified by comparing it to the stored instances (no rules have to be generated), the scope approach is instance-based. However, quite unconventionally, the scope approach:

- relies on partial pre-orderings $\leq_o$ between examples, indexed by objects. In particular, the number of neighbours of o that are kept is not fixed in the scope approach.
- has a rule-based characterisation: the notion of closeness to o that is used characterises the classes predicted by all the relevant and consistent classification rules for T that cover o.

The scope approach achieves an interesting trade-off between accuracy and efficiency. First, scope classification usually considers more neighbours than standard k-nearest neighbours. This makes it less sensitive to noise than these techniques. The price to be paid is a higher but still tractable time complexity (quadratic in the number of examples in the worst case). Second, since every relevant and consistent rule for T covering o is associated with a neighbour of o w.r.t. T in the scope approach, scope classification can prove more accurate than techniques where only a few classification rules are induced. Since rules do not have to be explicitly generated, it can be practical in situations where the number of rules that are relevant and consistent for T prevents them from being exhaustively computed.

The rest of this paper is organised as follows. The scope approach to classification is presented in Section 2. Its performance is compared with standard IBL and rule-based algorithms on many standard benchmarks from many domains in Section 3. Somer related research is discussed in section 4. The conclusions of our study are drawn in Section 5.

2 Scope Classification

In this section, the rule-based characterisation of the scope approach to classification is formally established. The scope algorithm is then presented and its efficiency is analysed. Finally, we show how the logical biases considered in the scope approach can be relaxed to allow it to deal with real-world (noisy) data.

2.1 Basics of the scope approach

Let us introduce the basic definitions of scope classification through an example. Let us consider a set of patients who suffer or not from a disease (y). Each individual is described by its sex s (Man or Woman), weight w in kilos and appetite a (Good, Average, Little).

Patient	s	w	a	y
e_1	M	70	G	T
e_2	W	55	A	F
e_3	M	50	L	T

Given a patient o described by $(s = M) \wedge (w = 75) \wedge (a = A)$, the aim of the classifier is to help the physician to detect whether o suffers or not from disease y. In this case, the classifier has to suggest a class for o given o and the training set $T = \{e_1, e_2, e_3\}$.

We are interested in the relevant and consistent rules for T covering o. Let R be such a rule. If R is relevant for T, then there exists at least one example e_R that satisfies R. Let us assume that $e_R = e_1$. Let us consider $R(o, e_1)$ the most specific rule covering o and e_1.

Definition 1. Let $C(o_1, o_2)$ be the least general generalisation of o_1 and o_2, i.e., $C(o_1, o_2)$ denotes the smallest hyper-rectangle containing the objects. $C(o_1, o_2)$ is defined as the conjunction of conditions (selectors) $C_i(o_1, o_2)$ on each attribute i:

- if the value of attribute i is missing in o_1 or in o_2 then $C_i(o_1, o_2) = True$,
- if i is nominal, if $v_i^{o_1} = v_i^{o_2}$ then $C_i(o_1, o_2) = (a = v_i^{o_1})$ else $C_i(o_1, o_2) = True$,
- if i is numerical or ordered, $C_i(o_1, o_2) = [min(v_i^{o_1}, v_i^{o_2}), max(v_i^{o_1}, v_i^{o_2})]$.

where attribute i is valued $v_i^{o_1}$ in o_1 and $v_i^{o_2}$ in o_2.

Let o be an object and e be an example labelled with $class(e)$. The most specific rule covering o and e is:

$$R(o, e) = C(o, e) \rightarrow (y = class(e))$$

For instance, $C(o, e_1) = (s = M) \wedge (w \in [70; 75]) \wedge (a \in [A; G])$.

By definition, every rule $R = c \rightarrow (y = class(e_1))$ covering o and e_1 is more general than $R(o, e_1)$. Hence, if R is consistent for T then $R(o, e_1)$ is consistent for T. To check whether $R(o, e_1)$ is consistent for T, we just have to check whether no example labelled with another class satisfies $C(o, e_1)$. In order to state it formally, let us define a partial pre-ordering $\leq_o$ between objects:

Definition 2. For every object o, let $\leq_o$ denote the partial pre-ordering defined by, for every o_1 and o_2, o_1 is closer to o than o_2, denoted $o_1 \leq_o o_2$, if and only if o_1 satisfies $C(o, o_2)$, i.e., $o_1 \models C(o, o_2)$.

The rule $R(o, e)$ is consistent for T if and only if no example labelled with a class different from $class(e)$ is closer to o than e according to $\leq_o$. Formally, let us define the neighbours of o w.r.t. T as:

Definition 3. Let e be an example of T and o be an object. Let $CE(e, T)$ denote the set of examples e' of T s.t. $class(e') \neq class(e)$. e is a neighbour of o w.r.t. T, denoted $e \in cons(T, \leq_o)$, if and only if $\forall e' \in CE(e, T), e' \nleq_o e$.

Stepping back to our example, e_2 does not satisfy $C(o, e_1)$, so $e_2 \nleq_o e_1$. Since $CE(e_1, T) = \{e_2\}$, e_1 is a neighbour of o w.r.t. T; in other words, the rule $R(o, e_1)$ is consistent and (obviously) relevant for T.

Let us emphasise that e is not required to be closer to o than all examples labelled with a class different from $class(e)$ to belong to $cons(T, \leq_o)$. What is needed is that no "counter-example" ce is closer to o than e. This means that either e is closer to o than ce, or that e and ce are not comparable w.r.t. $\leq_o$, that is the most frequent case. For instance, e_1 is not closer to o than e_2, but e_2 is not closer to o than e_1 (e_1 and e_2 are not comparable).

Let us also stress the fact that the number of neighbours of an object o that are considered in the scope approach is not fixed. This contrasts with the k-nearest neighbours techniques. Moreover, the notion of neighbourhood within scope classification is less restrictive than the ones considered within standard distance-based approaches. For example, if e is as close as possible to o w.r.t. Hamming distance (or w.r.t Minkowski distance $L_q(e_1, e_2) = \sqrt[q]{\Sigma_a (v_a^{e_1} - v_a^{e_2})^q}$), then it belongs to $cons(T, \leq_o)$ but the converse is not necessarily true. Roughly, attributes are considered as *incomparable dimensions* in the scope approach while they are only viewed as *numbers* that can be added and averaged in distance-based approaches. Results on every "dimensions" are dealt with not numerically as in [Demiroz and Guvenir, 1997], but logically.

2.2 The rule-based characterisation of the scope approach

As an immediate consequence of the definition of $cons(T, \leq_o)$, each time an example labelled with v_y belongs to $cons(T, \leq_o)$, a relevant and consistent rule for T labelling o with class v_y exists. Furthermore, the converse is true as well. Thus, the scope approach has a rule-based characterisation:

Theorem 4. *Let T be a set of examples and o be an object. There exists a consistent and relevant rule $c \rightarrow (y = v_y)$ for T that covers o if and only if there exists e in $cons(T, \leq_o)$ such that $class(e) = v_y$, and $C(o, e) \models c$.*

Based on this theorem, the scope algorithm prevents relevant and consistent rules from being generated by computing $cons(T, \leq_o)$ instead. Accordingly, scope classification requires no learning phase and no special types of abstractions (like decision trees, rules) have to be derived. However, rules can be easily derived, on a as-needed basis: For every e in $cons(T, \leq_o)$, the rule $R(o, e) = (C(o, e) \rightarrow (y = class(e)))$ is a relevant and consistent rule for T.

In our running example, e_1 is a neighbour of o w.r.t. T. If required, the rule $(s = M) \wedge (w \in [70; 75]) \wedge (a \in [A; G]) \rightarrow (y = T)$ can be generated.

2.3 The scope algorithm

A naive algorithm for generating $cons(T, \leq_o)$ consists in checking for every e whether no example from another class is closer to o than e. This algorithm requires $O(d \times |T|^2)$ comparisons between values of attributes, where d is the number of attributes[1]. A more efficient algorithm can take advantage of the "divide and conquer" paradigm and computes $cons(E_1 \cup E_2, \leq_o)$ as $cons(cons(E_1, \leq_o) \cup cons(E_2, \leq_o), \leq_o)$. Unfortunately, the former set is only a proper subset of the latter in the general case. Hence, each element resulting from the "divide and conquer" search must be compared with the examples of T labelled with other classes.

$cons(T, \leq_o)$ may contain examples labelled with different classes. It could easily be used to return a probability distribution over all classes. Since we are interested in predicting one class only, a *resolution criterion* must be used in the general case. Many criteria can be considered, including user-defined criteria that may incorporate some domain knowledge into classification. When no domain knowledge is available, *simple majority voting* is used: $class(o)$ is computed as the class value occurring the most frequently in $cons(T, \leq_o)$; *quadratic majority voting* is a variant where each example e is weighted by the square length of $C(o, e)$.

Analytically, the space complexity of the search of $cons(T, \leq_o)$ is in $\Theta(d \times |T|)$ in every case. Its time complexity is in $O(d \times C(|T|))$, where $C(|T|)$ is the number of comparisons w.r.t. $\leq_o$ that are performed. In the worst case, $C(|T|) = \Theta(2 * |T|^2)$. Such a quadratic time complexity (in the worst case) is higher than the time complexity of the simplest distance-based IBL techniques ($O(d \times |T|)$). In the best case, $C(T)$ is linear in the number of examples of T. Since a part of the algorithm is based on the "divide and conquer" paradigm, this time complexity is expected to be in $O(d \times |T| \times log_2(|T|))$. We checked it empirically on several benchmarks and we also observed that the worst case situation occurred only rarely.

2.4 Tuning consistency, generality and relevance

Dealing with real data requires the logical basements of scope classification to be relaxed. A tuning of consistency and a tuning of generality inspired from [Sebag, 1996] are presented and a tuning of relevance is introduced.

Actually, a rule that covers a few counter-examples must not be systematically dropped. Hence, the consistency requirement must be relaxed in order for a rule to accept at most ϵ counter-examples.

[1] For any set E, $|E|$ denotes its cardinality.

Definition 5. Let $NI(e,T)$ be the number of examples ce s.t. $class(ce) \neq class(e)$ and ce is closer to o than e:

$$NI(e,T) = |\{ce \in CE(e,T)|ce \leq_o e\}|$$

An example e is an ϵ-neighbour of o w.r.t. T if and only if $NI(e,T) \leq \epsilon \times |T|$.

Some attributes may be of no interest in some part of the universe, thus forgetting some of them (at most M) can prove valuable.

Definition 6. An object o_1 is closer to an object o than an object o_2 except on (at most) M attributes, denoted $o_1 \leq_o^M o_2$, if and only if $|\{\text{attribute } i|(i = v_i^{o_1}) \not\models C_i(o, o_2)\}| < M$. An example e is an M-neighbour of o w.r.t. T, denoted $e \in M - cons(T, \leq_o)$, if and only if $e \in cons(T, \leq_o^M)$.

The relevance requirement can also be strengthened in order to only consider rules satisfied by at least γ examples.

Definition 7. Let $NC(e,T)$ be the number of examples f s.t. $class(f) = class(e)$ and f is closer to o than e:

$$NC(e,T) = |\{f \in T|class(f) = class(e) \text{ and } f \leq_o e\}|$$

An example e is an γ-neighbour of o w.r.t. T if and only if $NC(e,T) > \gamma \times |T|$.

These three parameters can be incorporated all together within the scope approach (the notion of neighbour becomes the notion of (ϵ, M, γ)-neighbour).[2] Their values and the resolution criterion are automatically assessed; we keep the values and criterion for which the accuracy of the corresponding classifier measured by a 10-fold cross-validation on a randomly chosen subset S of T[3] is maximal. Values of ϵ and γ range from 0% to 30% using an increment of 5% and M ranges from 0 to the number of attributes d. Simple and quadratic majority voting are considered as resolution criteria. This is analogous to the wrapper method of [Kohavi and John, 1995]. However, while parameters ϵ, M and γ and the resolution criterion should depend on the distribution of the training set, we assume they depend on the domain only. Thus they are only computed once for a given domain: they are not re-assessed when different Ts are considered over the same domain.

3 An Empirical Evaluation

The performance of scope classification has been compared to some usual instance-based and rule-based classifiers, in the empirical framework described below. Both accuracy and execution time have been considered.

[2] Note that the relation $\leq_o^M$ is no longer a pre-ordering since transitivity is lost. However this does not question the correctness of the scope algorithm.

[3] $|S| = 0.1 \times |T|$ whenever $|T| \geq 100$, $0.9 \times |T|$ otherwise.

3.1 The empirical framework

Experiments have been carried out to compare scope classification with some of the most famous rule-based or instance-based approaches to classification. Thus, PEBLS 3.0 [Cost and Salzberg, 1993], a state-of-the art IBL system has been used. Three rule-based learning algorithms have been considered: two of them generate only some rules, CN2 [Clark and Niblett, 1989] and C4.5 [Quinlan, 1993], while the third one, SE-Learn [Rymon, 1996a], builds up all relevant and consistent rules for the training set when possible. We also compare our approach empirically with RISE [Domingos, 1996], an approach unifying rule-based and instance-based learning. The default classifier (always choosing the most frequent class) has also been included in the study as a baseline.

While Kohavi and John [Kohavi and John, 1995] showed that an automatic assignment of parameters could entail a better accuracy (with the drawback of a longer training time), none of those programs includes it. We simply used the latest versions distributed by their authors, using default values except when some other values are known to give better results. In particular, the exemplar weighting "used_correct" as described in [Cost and Salzberg, 1993], with ten trials, was used for PEBLS. Default values of the latest version of CN2 (6.1) were used. C4.5 was used with rules generation and windowing (growing ten trees, the default), requiring a minimum of four examples (instead of two) in the two branches of a test, and using a confidence level of 37.5% (instead of 25%) for rule pruning. Simple majority voting has been chosen as a resolution criterion for both SE-trees. SE-Learn has been run without pruning and has also been run with significance-based statistical pruning at the $p < 0.05$ level.

We have compared the classification accuracies obtained by the classification techniques described above on many domains. The domains datasets used in our experiments have been drawn from the UCI repository [Merz and Murphy, 1996]: audiology (AD), annealing (AN), credit (CE), pima diabetes (DI), echocardiogram (EC), glass (GL), heart disease (Cleveland HDc, Hungarian HDh, Switzerland HDs and V.A. medical center HDv databases), hepatitis (HE), horse colic (HO), iris (IR), labor negotiation (LA), lung cancer (LC), liver disease (LD), contact lenses (LE), LED (LI), post-operative (PO), DNA promoters (PR), solar flares (common SFc, moderate SFm and severe SFx), soybean (SO), splice junctions (SP),voting records (VO), wine (WI) and zoology (ZO).

3.2 Accuracy and execution time

Accuracy is measured by 10-fold cross-validation. Table 1 reports accuracy and standard deviation measured for each dataset, and the confidence level in the hypothesis H_1 = "the difference of accuracy between this classifier and scope classification is significant" using a one-tailed paired t test.

Results of table 1 are summarized on table 2 according to five measures that can be used to compare classifiers:

- **Number of wins.** It counts the number of datasets where scope classification achieves higher accuracy than the other algorithm and those where the

Table 1. Average accuracies and standard deviations. Superscripts denote confidence level: 1 is 99.5%, 2 is 99%, 3 is 97.5%, 4 is 95%, 5 is 90% and 6 is below 90%. - stands for datasets where SE-Learn has been interrupted (more than one CPU day or too much memory was required).

Dom	SCOPE	RISE	PEBLS	C4.5	CN2	SE	SE 0.05	Default
AD	66.5 ± 10.5	80.0 ± 11.1^{1}	71.8 ± 0.1^{1}	70.3 ± 10.4^{5}	77.0 ± 12.1^{1}	-	-	18.0 ± 9.8^{1}
AN	94.2 ± 2.7	98.4 ± 1.6^{1}	99.2 ± 0.0^{1}	93.2 ± 1.7^{6}	85.7 ± 3.7^{1}	-	-	76.2 ± 4.0^{1}
CE	86.2 ± 4.9	82.2 ± 5.0^{1}	83.3 ± 0.1^{3}	83.7 ± 5.1^{2}	82.6 ± 4.7^{1}	86.3 ± 4.5^{6}	89.0 ± 5.1^{3}	55.5 ± 4.7^{1}
DI	74.7 ± 4.7	70.3 ± 3.4^{3}	73.8 ± 0.0^{6}	72.9 ± 4.6^{6}	73.8 ± 3.1^{6}	72.7 ± 4.3^{6}	75.9 ± 2.4^{6}	65.1 ± 5.6^{1}
EC	69.3 ± 13.0	62.7 ± 11.9^{6}	63.7 ± 0.2^{6}	68.0 ± 12.4^{6}	70.1 ± 10.1^{6}	67.8 ± 11.5^{6}	68.5 ± 13.5^{6}	67.1 ± 14.7^{4}
GL	74.3 ± 13.7	72.0 ± 7.6^{6}	69.7 ± 0.1^{4}	63.3 ± 7.0^{2}	62.1 ± 13.1^{1}	71.4 ± 10.6^{6}	58.2 ± 10.5^{1}	30.3 ± 13.9^{1}
HDc	83.8 ± 4.5	79.2 ± 6.5^{1}	80.1 ± 0.1^{5}	74.9 ± 3.8^{1}	77.2 ± 6.1^{3}	82.8 ± 5.6^{6}	88.7 ± 5.3^{1}	54.0 ± 7.7^{1}
HDh	82.9 ± 4.4	76.4 ± 8.0^{2}	77.3 ± 0.1^{1}	78.7 ± 5.0^{1}	75.1 ± 6.3^{1}	81.6 ± 7.0^{6}	86.6 ± 6.7^{2}	64.1 ± 10.1^{1}
HDs	93.5 ± 3.3	92.6 ± 4.7^{6}	90.9 ± 0.1^{5}	90.9 ± 2.0^{5}	92.6 ± 2.6^{6}	93.5 ± 3.5^{6}	82.1 ± 19.0^{4}	93.5 ± 3.4^{2}
HDv	75.9 ± 11.4	71.4 ± 9.0^{5}	64.2 ± 0.1^{1}	75.2 ± 12.9^{6}	74.8 ± 11.1^{6}	75.3 ± 12.7^{6}	79.0 ± 13.2^{6}	74.4 ± 11.2^{5}
HE	78.7 ± 13.3	76.9 ± 14.6^{6}	82.4 ± 0.1^{6}	81.6 ± 10.1^{6}	81.2 ± 11.1^{6}	-	-	79.4 ± 14.3^{6}
HO	82.1 ± 6.1	82.6 ± 6.0^{6}	82.5 ± 0.0^{6}	84.3 ± 5.3^{6}	82.1 ± 6.5^{6}	-	-	63.1 ± 8.1^{1}
IR	94.7 ± 5.8	95.3 ± 7.1^{6}	95.5 ± 0.1^{6}	95.8 ± 5.5^{6}	94.0 ± 5.8^{6}	96.0 ± 5.6^{6}	90.6 ± 7.2^{3}	21.3 ± 5.3^{1}
LA	87.3 ± 14.4	87.3 ± 23.0^{6}	86.8 ± 0.2^{6}	78.4 ± 10.8^{5}	78.7 ± 11.4^{5}	89.0 ± 20.5^{6}	83.0 ± 33.3^{6}	64.3 ± 29.4^{1}
LC	48.3 ± 28.1	43.3 ± 32.8^{6}	42.3 ± 0.2^{6}	59.9 ± 21.1^{6}	44.2 ± 16.7^{6}	-	-	40.8 ± 15.0^{6}
LD	75.8 ± 8.1	75.8 ± 4.9^{6}	71.4 ± 0.1^{5}	74.8 ± 5.6^{6}	78.2 ± 6.0^{6}	78.5 ± 8.5^{6}	81.5 ± 7.8^{3}	74.7 ± 8.8^{6}
LE	70.0 ± 26.7	78.3 ± 29.5^{6}	72.0 ± 0.3^{6}	87.7 ± 19.0^{4}	70.0 ± 28.1^{6}	70.0 ± 28.1^{6}	93.3 ± 21.1^{3}	63.3 ± 27.0^{6}
LI	59.0 ± 15.8	55.0 ± 13.5^{5}	51.6 ± 0.1^{5}	55.7 ± 8.5^{6}	63.0 ± 14.2^{6}	67.1 ± 23.2^{6}	60.4 ± 17.4^{6}	18.0 ± 14.0^{1}
PO	71.1 ± 11.3	63.3 ± 21.6^{6}	57.0 ± 0.2^{1}	67.9 ± 11.0^{4}	63.3 ± 14.9^{4}	63.8 ± 14.1^{1}	26.7 ± 14.1^{1}	71.1 ± 11.9^{1}
PR	82.0 ± 14.4	84.0 ± 8.6^{6}	87.5 ± 0.1^{5}	84.3 ± 9.7^{6}	70.8 ± 7.7^{3}	-	-	37.7 ± 5.9^{1}
SFc	88.9 ± 3.2	89.2 ± 3.4^{6}	83.0 ± 0.0^{1}	87.9 ± 3.4^{1}	88.8 ± 3.4^{6}	86.8 ± 4.4^{4}	88.4 ± 8.3^{6}	88.8 ± 3.4^{6}
SFm	90.1 ± 5.9	89.1 ± 5.8^{5}	83.5 ± 0.1^{1}	88.4 ± 6.1^{3}	89.1 ± 6.8^{6}	89.1 ± 6.7^{6}	92.2 ± 6.2^{5}	90.1 ± 6.2^{6}
SFx	97.8 ± 2.4	97.8 ± 2.5^{3}	96.2 ± 0.0^{1}	97.8 ± 2.5^{3}	97.5 ± 2.8^{6}	97.4 ± 3.0^{6}	47.1 ± 35.6^{1}	97.8 ± 2.5^{3}
SO	100.0 ± 0.0	100.0 ± 0.0^{6}	100.0 ± 0.0^{6}	96.8 ± 7.4^{6}	97.5 ± 7.9^{6}	100.0 ± 0.0^{6}	100.0 ± 0.0^{6}	37.5 ± 26.4^{1}
SP	95.3 ± 1.5	92.7 ± 1.3^{1}	94.0 ± 0.0^{3}	93.7 ± 1.4^{1}	91.2 ± 2.0^{1}	-	-	51.9 ± 2.4^{1}
VO	93.6 ± 4.2	95.9 ± 3.2^{4}	94.8 ± 0.0^{5}	95.1 ± 3.1^{6}	94.7 ± 3.3^{6}	-	-	61.4 ± 7.0^{1}
WI	95.5 ± 4.3	98.9 ± 2.4^{4}	97.7 ± 0.0^{4}	94.9 ± 3.7^{6}	92.7 ± 4.6^{6}	98.8 ± 3.7^{3}	96.5 ± 4.1^{6}	39.8 ± 10.1^{1}
ZO	94.0 ± 9.2	96.0 ± 7.0^{6}	95.5 ± 0.1^{6}	88.6 ± 14.0^{3}	90.0 ± 15.6^{5}	95.0 ± 9.7^{6}	95.8 ± 6.8^{6}	40.4 ± 16.0^{1}

converse happens (draws are not counted either way). For instance, scope algorithm performed better than RISE in 14 datasets and worse in 10.

- **Number of significant wins.** It only counts a dataset when the confidence level in the difference of accuracy is greater than 95%.
- **Wilcoxon test.** This is a non-parametric approach to paired t test. We give the confidence level in hypothesis H_1.
- **Average.** It reports the average accuracy over all domains.
- **Ranks.** For each dataset, accuracies are ranked and are given values from 0 (the worst one) to 1 (the best one) in a uniform way. The global rank is averaged over all domains.

Experiments show that SE-Learn has a better accuracy than scope classification which has a better accuracy than RISE. These three algorithms have better accuracies than PEBLS, C4.5, CN2 and the default classifier.

We have also compared the efficiencies of both approaches. Results are summarized on table 3. In all the experiments, the execution time is the time required by each technique to complete the 10-fold cross-validation from scratch. Thus, for the scope approach, it is the time required to assess the parameters plus the time required to classify the ten test sets (parameters are only assessed once for the ten test sets). For the other approaches, the execution time is the learning time plus the classification time.

Table 2. Comparison of accuracies according to five measures.

Measure	SCOPE	RISE	PEBLS	C4.5	CN2	SE	SE 0.05	Default
No. wins	-	14-10	17-10	18-8	19-7	10-7	6-11	23-1
No. signif. wins	-	5-4	9-3	9-1	8-1	2-1	3-6	18-0
Wilcoxon test	-	80	98	92.5	99.5	<80	80	100
Average	82.3	81.7	80.3	81.6	79.9	83.1	83.9	58.6
Rank	0.67	0.58	0.47	0.55	0.47	0.69	0.75	0.20

Table 3. Comparison of execution times according to three measures.

Measure	SCOPE	RISE	PEBLS	C4.5	CN2	SE	SE 0.05
No. wins	-	19-9	3-25	11-17	6-22	10-10	2-18
Wilcoxon test	-	95.5	100	89	100	<80	99.9
Rank	0.34	0.21	0.79	0.52	0.59	0.41	0.67

We used the following measures:

- **Number of wins.** It counts the number of datasets where scope classification had a smaller execution time than the other algorithm and those where the converse happens (draws are not counted either way).
- **Wilcoxon test.** It is the confidence level in hypothesis H_1 = "the difference of execution time between this classifier and scope classification is significant".
- **Ranks.** For each dataset, execution times are ranked and are given values from 0 (the worst one) to 1 (the best one) in a uniform way. The global rank is averaged over all domains.

We can observe that the execution time of scope classification measured during this evaluation is on average smaller than the one of RISE, but greater than those of PEBLS, C4.5 and CN2. On some benchmarks, the execution time of SE-Learn without pruning is similar to the one of scope classification, and with pruning, it is quite better. However, all the relevant and consistent rules for T must be considered in SE-Learn. Since their number is exponential in the number of attributes d in the worst case, the time required by SE-Learn on a classification task can be much higher than those of the other approaches. For example, SE-Learn had to be interrupted in 8 cases out of 28, namely whenever the number of attributes exceeds 16 (unless the number of examples N was very small).

4 Related Research

In this section, the scope approach is shown to closely relate to approaches where the logical biases of relevance and consistency are considered, in particular the disjunctive version space approach [Sebag, 1996] and the SE-Learn framework [Rymon, 1996a]. Differences between the scope approach and previous approaches combining IBL and rule-based learning are emphasized.

4.1 The disjunctive version space approach

The disjunctive version space ($DiVS$ for short) of T is the disjunction of the version spaces $H(e)$ for every example $e \in T$. The version space $H(e)$ of e is the conjunction of the hypotheses $D(e, ce)$ that discriminate e from its counter-examples $ce \in CE(e,T)$. $D(e, ce)$ is the disjunction of the maximally discriminant selectors $SEL_i(e, ce)$ for each attribute i.

In $DiVS$, an example e is a (ϵ, M)-neighbour of an object o w.r.t. T if and only if

$$|\{ce \in CE(e,T) | |\{i | v_i^o \in SEL_i(e, ce)\}| \not> M\}| \leq \epsilon \times |T|.$$

Stepping back to the patient example, $H(e_1) = D(e_1, e_2) = (w > 55) \vee (a > A)$. This hypothesis is clearly different from the left-hand sides of rules considered in scope classification: $C(o, e_1) = (s = M) \wedge (w \in [70; 75]) \wedge (a \in [A; G])$. Whereas the M parameter seems different, the (ϵ, M)-neighbourhoods of an object coincide in both approaches:

Theorem 8. *Let o be an object and e an example from T.*
$o \in (\epsilon, M) - H(e)$ if and only if $e \in (\epsilon, M) - cons(T, \leq_o)$.

Thus, for the same resolution criterion and parameters ϵ and M, scope classification and the $DiVS$ approach lead to the same neighbourhood. The generality parameter M can be considered from at least two points of view. In scope classification, a counter-example ce may satisfy a rule $R(o, e)$ except on at most M attributes (i.e., $ce \leq_o^M e$). In $DiVS$, an object o must satisfy at least M selectors $D(e, ce)$ discriminating ce from e.

A parameter similar to γ could be used in $DiVS$, but since the hypotheses considered in $DiVS$ (a version space) and in the scope classification (a rule) differ, the neighbourhoods are no longer equivalent. For instance, if we consider the following example in R^2: $o(0; 0)$ an object, $e_1(1; 1)$ and $e_2(-1; 1)$ two positive examples, and $ce_1(1; 2)$ and $ce_2(2, -1)$ two negative examples then $e_2 \in H(e_1)$ but $e_2 \notin C(o, e_1)$. Thus an example is more likely to be kept in $DiVS$ than in the scope classification. Finally, since it combines IBL with rule-based learning, scope classification allows for extensions that cannot be envisioned in the current version of $DiVS$; for instance, examples could be easily generalized into rules in a learning phase within the scope approach.

4.2 SE-Learn

SE-Learn is a rule-based approach based on the same logical biases of relevance and consistency used in the scope approach. It generates all the (most general) relevant and consistent rules for T. If no statistical bias (including the resolution criterion) were used, scope classification and SE-Learn would be closely related, according to Theorem 4. Indeed, for every object o, there exists a (most general) relevant and consistent rule R for T that covers o iff there exists an example e in T s.t. $e \in cons(T, \leq_o)$ and the class value of e is the class value of the right-hand side of R. But there is no quantitative side in Theorem 4. Thus, the number of most general rules that are relevant and consistent for T and cover an object o may easily differ from the number of neighbours of o w.r.t. T in the scope approach. Accordingly, equipped with the same resolution criterion, the two approaches do not give rise to the same classifiers. Moreover, the other statistical bias used in both approaches (parameters ϵ, M, γ in scope classification and statistical pruning p in SE-Learn) do not coincide. Finally, scope classification is an IBL technique while SE-Learn is a rule-based one. As mentioned in Section 3.2, the computational complexity of SE-Learn makes it impractical for problems with many attributes and examples.

4.3 Other related works

Several approaches combine instance-based and rule-based learning, including NGE [Salzberg, 1991], BNGE [Wettschereck and Dietterich, 1995] and RISE [Domingos, 1996]. These approaches generalize the examples from the training set into rules in a learning phase, then classify every object according to its closest rule (w.r.t. some distance). Thus, the rules used to classify an object o do not depend on the object itself (they are fixed during the learning phase). In any case, only one rule is elected to classify o; for instance, the most specific rule among the closest to o [Salzberg, 1991], or the one with the best Laplace accuracy [Domingos, 1996].

Clearly enough, these approaches are very different from scope classification. First, they use rules as instances, while no rules have to be generated within the scope approach (no learning phase is mandatory). Second, they are distance-based while scope classification is not. Third, all the examples e of the training set s.t. $R(o, e)$ is consistent for T are considered in scope classification when o is to be classified. Such rules $R(o, e)$ are different from those considered in the approaches mentioned above in the general case. In particular, they depend on o (every $R(o, e)$ must cover o). Finally, the resolution criteria used in all these approaches differ.

5 Conclusion

The scope approach is an IBL technique with a rule-based characterization. Since it is a bottom-up approach, continuous attributes do not need to be discretized within scope classification. Since the scope algorithm does not focus on a fixed number of neighbours, it appears empirically more accurate than PEBLS where the number of neighbours is constrained. Since every rule associated with a neighbour is implicitly considered in the scope approach, it appears empirically as more accurate than techniques where only a few classification rules are induced, in particular decision trees (C4.5) and CN2. While the whole set of (most general) relevant and consistent rules for T is often too huge to be computed explicitly, the scope approach does not require this set to be generated. Hence it can be practical in many situations where SE-Learn is not.

Acknowledgements

The authors would like to thank Michèle Sebag and anonymous reviewers for their numerous suggestions of improvements, Ron Rymon and Bob Schrag for providing them with the SE-Learn program and for their help and comments. Thanks also to the authors of CN2, PEBLS, C4.5 and RISE systems, and to datasets providers for the UCI repository. The first author is currently supported by the Esprit Long Term Research Project 20237 (ILP2). The second author is supported in part by a "*contrat d'objectif de la région Nord/Pas-de-Calais*".

References

[Clark and Niblett, 1989] Peter Clark and Tim Niblett. The CN2 induction algorithm. *Machine Learning*, 3(4):261–283, 1989.

[Cost and Salzberg, 1993] Scott Cost and Steven Salzberg. A weighted nearest neighbor algorithm for learning with symbolic features. *Machine Learning*, 10:57–78, 1993.

[Demiroz and Guvenir, 1997] Gulsen Demiroz and H. Altay Guvenir. Classification by voting feature intervals. In *Proc. of the Ninth European Conference on Machine Learning*, pages 85–92, Prague, Czech Republic, 1997. LNAI 1224, Springer-Verlag.

[Domingos, 1996] Pedro Domingos. Unifying instance-based and rule-based induction. *Machine Learning*, 24:141–168, 1996.

[Kohavi and John, 1995] Ron Kohavi and George H. John. Automatic parameter selection by minimizing estimated error. In *Proc. of the Twelfth International Conference on Machine Learning*, 1995. Morgan Kaufmann.

[Merz and Murphy, 1996] C.J. Merz and P.M. Murphy. UCI repository of machine learning databases. http://www.ics.uci.edu/~mlearn/MLRepository.html, 1996.

[Quinlan, 1993] John Ross Quinlan. *C4.5 : Programs for machine learning*. Series in machine learning. Morgan Kaufmann, 1993.

[Rymon, 1993] Ron Rymon. An SE-tree based characterization of the induction problem. In *Proc. of the Tenth International Conference on Machine Learning*, pages 268–275, 1993.

[Rymon, 1996a] Ron Rymon. SE-Learn home page. http://www.isp.pitt.edu/~rymon/SE-Learn.html, 1996.

[Rymon, 1996b] Ron Rymon. SE-trees outperform decision trees in noisy domains. In *Proc. of the Second International Conference on Knowledge Discovery in Databases*, pages 331–334, 1996. AAAI Press.

[Salzberg, 1991] Steven Salzberg. A nearest hyperrectangle learning method. *Machine Learning*, 6:251–276, 1991.

[Sebag, 1996] Michèle Sebag. Delaying the choice of bias: A disjunctive version space approach. In *Proc. of the Thirteenth International Conference on Machine Learning*, pages 444–452, 1996. Morgan Kaufmann.

[Wettschereck and Dietterich, 1995] Dietrich Wettschereck and Thomas G. Dietterich. An experimental comparison of the nearest-neighbor and nearest-hyperrectangle algorithms. *Machine Learning*, 19:5–27, 1995.